第 37 卷第 5 期
录用定稿
计算机应用研究
Application Research of Computers
Vol. 37 No. 5
Accepted Paper

基于特征点法和直接法 VSLAM 的研究现状 *

邹　雄[1]，肖长诗[1,2]，文元桥[1,2,3]，元海文[1]

(1. 武汉理工大学 航运学院，武汉 430063; 2. 内河航运技术湖北省重点实验室，武汉 430063; 3. 国家水运安全工程技术研究中心，武汉 430063)

摘　要：基于视觉的同时定位和建图（VSLAM）分为前端和后端，前端包括视觉里程计和回环检测，后端包括后端优化和建图。按照估计相机运动的不同方式，将 VSLAM 分为特征点法和直接法。首先从这两个方面对前端进行综述，阐述其中的关键技术和最新的研究进展，对比分析不同方法的优缺点；然后详细分析优化后端与滤波器后端的区别，进一步地对多个开源代码进行比较研究，分析它们的优劣势和适用场合；再讨论深度学习、语义地图和多机器人在 VSLAM 领域的研究进展，以及相关技术与 VSLAM 的结合方式及前景；最后对 VSLAM 的未来进行展望。

关键词：VSLAM；视觉里程计；特征点法；直接法；非线性优化

中图分类号：TP391.4　　doi: 10.19734/j.issn.1001-3695.2018.11.0789

Current research of feature-based and direct methods VSLAM

Zou Xiong[1], Xiao Changshi[1, 2], Wen Yuanqiao[1, 2, 3], Yuan Haiwen[1]

(1. *School of Navigation*, *Wuhan University of Technology*, *Wuhan* 430063, *China*; 2. *Hubei Inland Shipping Technology Key Laboratory*, *Wuhan* 430063, *China*; 3. *National Engineering Research Center for Water Transport Safety*, *Wuhan* 430063, *China*)

Abstract: Visual simultaneous localization and mapping (VSLAM) is divided into front-end and back-end. The front-end includes visual odometry and loop detection, and the back-end includes back-end optimization and mapping. This paper divided VSLAM into feature-based method and direct method according to different ways of estimating camera motion. Firstly, it summarized the front-end from these two aspects, the key technologies and elaborated the latest research progress, and compared and analyzed different methods. Then, it analyzed the differences between the optimize back-end and the filter back-end in detail. Further, it compared the advantages and disadvantages of several open source codes and their applicable occasions. Then, it introduced the research progress of deep learning, semantic mapping and multi-robots in VSLAM, and discussed the combination of related technologies with VSLAM and its prospects. Finally, it prospected the future of VSLAM.

Key words: VSLAM; VO; feature-based method; direct method; nonlinear optimization

同时定位与地图构建（simultaneous localization and mapping，SLAM）[1,2]是机器人进入未知环境遇到的第一个问题。它是指机器人搭载特定传感器，在没有环境先验信息的情况下，于运动过程中对周围环境建模并同时估计自身的位姿[3]。如果传感器主要为相机，那么就称为视觉 SLAM（VSLAM）[4]。SLAM 技术已经研究和发展了三十多年，研究人员已经做了大量的工作，近十年来，随着计算机视觉的发展，VSLAM 以其硬件成本低廉、轻便、高精度等优势获得了学术界和工业界的青睐。

1　VSLAM 的系统框架

VSLAM 是利用多视图几何理论[5]，根据相机拍摄的图像信息对相机进行定位并同时构建周围环境地图。按照相机的分类，有单目、双目、RGBD、鱼眼、全景等。为了方便，本文只考虑普通相机。

从 VSLAM 的提出到目前为止，经过研究者们十多年不懈努力，VSLAM 框架基本形成，如图 1 所示。VSLAM 主要包括视觉里程计（visual odometry，VO）、后端优化、回环检测、建图。其中 VO 研究图像帧间变换关系完成实时的位姿跟踪，对输入的图像进行处理，计算姿态变化，得到相机间的运动关系；但是随着时间的累计，误差会累积，这是由于仅仅估计两个图像间的运动造成的，后端主要是使用优化方法，减小整个框架误差（包括相机位姿和空间地图点）。回环检测，又称闭环检测，主要是利用图像间的相似性来判断是否到达过先前的位置，以此来消除累计误差，得到全局一致性轨迹和地图。建图，根据估计的轨迹，建立与任务要求对应的地图。

现在比较通常的惯例是把 VSLAM 分为前端和后端。前端为视觉里程计和回环检测，相当于是对图像数据进行关联；后端是对前端输出的结果进行优化，利用滤波或非线性优化理论，得到最优的位姿估计和全局一致性地图。

2　前端

2.1　视觉里程计

前端中的视觉里程计，为通过采集的图像得到相机间的运动估计，视觉里程计问题可由图 2 描述（双目立体视觉里

收稿日期：2018-11-10；**修回日期：**2019-01-17　**基金项目：**国家自然科学基金资助项目（51579204，51679180）；武汉理工大学自主创新研究基金项目（2016IVA064,2016-YB-029）

作者简介：邹雄（1982-），男，博士研究生，主要研究方向为机器视觉（zx2000@whut.edu.cn）；肖长诗（1974-），教授，博士，主要研究方向为宽动态成像；文元桥（1974-），教授，博士，主要研究方向为大数据；元海文（1988-），博士研究生，主要研究方向为机器视觉.

程计）。视觉系统在运动过程中，在不同时刻获取了环境的图像，而且相邻时刻的图像必须有足够的重叠区域，则视觉系统的相对旋转和平移运动可被估算出来，然后将每两个相邻时刻之间视觉系统的运动串联起来,可以得到累计的视觉系统相对于参考坐标系的旋转和平移。

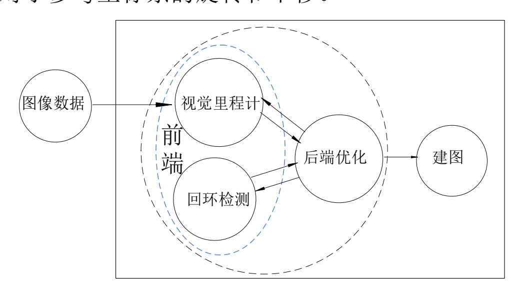


图 1　VSLAM 系统框架

Fig. 1　VSLAM system framework

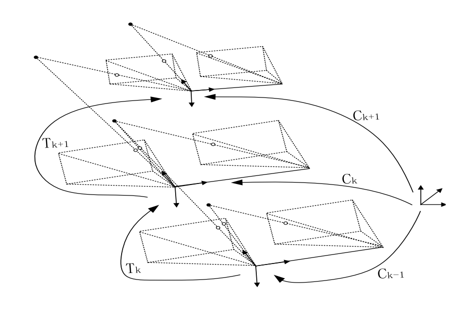


图 2　视觉里程计的问题描述示意图

Fig. 2　VO problem description diagram

如图 2 所示，视觉里程计的任务就是已知 $k=0$ 的初始位置 C_0 （这可以根据情况自己定义），求相机的运动轨迹 $C_{0:n}=\{C_0,...,C_n\}$ ，即当前的位置 C_k 通过 T_k 和上一时刻的位置 C_{k-1} 来计算，算式为 $C_k=C_{k-1}*T_k$ 。其中： T_k 为 K 和 K+1 时刻的相机相对位置变化,可根据相应时刻采集的图像计算出来，从而恢复相机的运动轨迹。

视觉里程计可分为特征点法和直接法（图 3）。特征点法主要是根据图像上的特征匹配关系得到相邻帧间的相机运动估计，它需要对特征进行提取和匹配，然后根据匹配特征构建重投影误差函数,并将其最小化从而得到相机的相对运动;直接法是假设两帧图像中的匹配像素的灰度值不变，构建光度误差函数，也将其最小化求解帧间的相机运动。

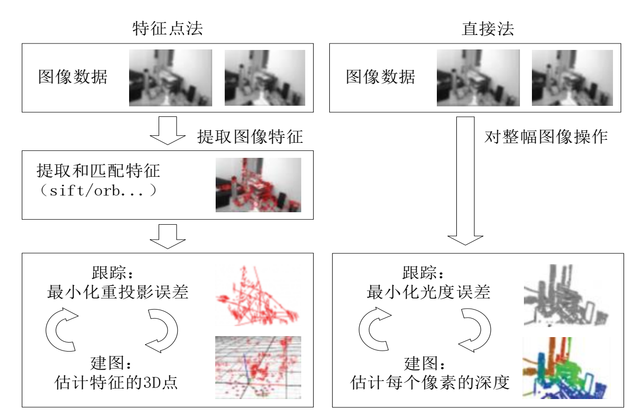


图 3　特征点法和直接法 VSLAM 系统示意图

Fig. 3　Schematic diagram of feature-based method and direct method VSLAM system

2.1.1 特征点法

特征点法的原理为：通过提取和匹配相邻图像的特征点估计该帧间对应的相机相对运动。特征点法的步骤包括特征检测、匹配、运动估计和优化，如图 4 所示。

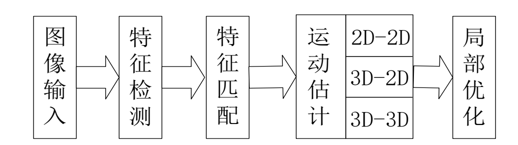


图 4　特征点法流程示意图

Fig. 4　Flow diagram of feature-based method

特征点可以称为兴趣点、显著点、关键点等。以点的位置来表示的点特征是一种最简单的图像特征。特征点可以分为关键点和描述子两部分。事实上，特征点是一个具有一定特征的局部区域的位置标志，称其为点，是将其抽象为一个位置概念,以便于确定两幅图像中同一个位置点的对应关系,所以在特征匹配过程中是以该特征点为中心，将邻域的局部特征进行匹配。也就是说在进行特征匹配时首先要为这些特征点建立特征描述，这种特征描述通常称之为描述子。本文希望特征点在不同时刻不同位置都能保持稳定，一个好的特征点应该拥有可重复性、可区别性、高效性。

VSLAM 中常用的特征检测算法主要有 SIFT[6,7]、SURF[8]、FAST[9]、ORB[10]等。每种都有自己的优劣[11]，其中尺度不变特征转换(scale-invariant feature transform，SIFT)首先通过用差分高斯（DoG）算子对图像的上下尺度进行卷积运算，然后在尺度和空间上获取输出的局部最小值或最大值。SURF 建立在 SIFT 上，也叫 SIFT 加速版，它使用盒式滤波器来近似高斯滤波器，它们充分考虑了在图像变换过程中出现的光照、尺度、旋转等变化，从这点上看非常适合 SLAM，但随之而来的是极大的计算量。到目前为止，如果实时地利用 SIFT 特征进行 VSLAM，还需要 GPU 加速。

FAST 是一种角点，主要检测局部像素灰度变化明显的地方。如果候选关键点像素灰度值与邻域的像素灰度值差别过大（比如邻域采用半径为 3 的圆上连续像素点超过 9），那么它即为角点。FAST 的特点是快，但是它不具备尺度和旋转的不变性。

ORB 对原始的 FAST 算法进行了改进：对原始的 FAST 角点分别计算 Harris 响应值，然后排序和选取较大响应值的角点；通过构建图像金字塔降采样，并在每一层上检测角点实现尺度不变特性；以图像块的灰度质心和几何中心得到特征点的方向。不仅如此，ORB 在提取 FAST 角点后还使用了 BRIEF 特征描述。

BRIEF[12]是一种二进制编码的特征描述子，它使用从关键点周围的块中采样的成对亮度比较。由于使用二进制表达和存储，所以速度非常快。原始的 BRIEF 描述子没有考虑方向，而 ORB 在提取 FAST 角点时考虑了尺度和方向，所以 ORB 即具备了 FAST 和 BRIEF 的速度快的特点,又具备了较好的尺度和旋转不变性。

早期特征点的匹配多采取跟踪方式，比如检测关键点(不需要描述子)，采用光流跟踪得到关键点的匹配。通常为了排除误跟踪，可以采用一致性检测。这种适合相邻帧之间的运动量和外观变化较小的情况。

如果两帧之间的运动量和外观变化较大，本文需要计算两帧之间的特征点和描述子，比较描述子间的距离（如汉明距离）。由于计算量的关系，很少采用穷尽的方式进行匹配，多采用恒速等模型在预期区域中搜索潜在的对应关系。如果是双目匹配或者深度滤波器中计算每个像素的深度，通常采用极线搜索和采用归一化互相关（normalized cross correlation，NCC）或绝对误差和（sum of squared differences，SSD）找

到匹配点。对于双目来说，为了保证准确匹配，可以采用环形检测对左右和前后总共四张图像验证是否形成匹配环[13]。

运动估计就是根据特征点的匹配情况，恢复出两帧间的相机运动。针对特征点匹配的情况，运动估计分为 2D-2D、3D-2D、3D-3D（图 5）。其求解方法可以分为几何方法和优化方法。几何方法主要是根据对极几何理论得到两帧间的对应关系；而优化方法主要是构建两帧间的重投影误差并使其最小，从而得到帧间变换。

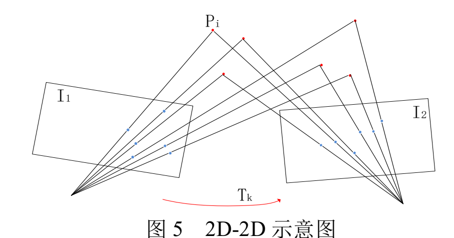


图 5　2D-2D 示意图

Fig. 5　2D-2D schematic diagram

2D-2D 主要是针对单目相机的初始化过程，在不知道空间中 3D 点的情况下（如未进行初始化）通过两帧间匹配的特征点进行帧间相机运动估计（图 5）。这涉及到对极几何中本质矩阵（E）或单应性矩阵（H）的相关理论及其分解，通常在图像的特征匹配中难免会有“外点”，可以采用随机采样一致（RANSAC）得到最大“内点”子集的 E 或 H。对极约束（图 6），P_1、P_2 和 t 共面得到 $p_2^T \cdot (t \times p_1^{'}) = 0$，进一步得到 $p_2^T E p_1 = 0$，其中 $E = [t]_{\times} R$。针对 E 的分解，经典的八点法是当作线性方程来解[14]，然后把结果投影到 E 所在的流形上（利用 E 的内在性质[15]）。另一方面，E 有五个自由度最小可以通过 5 点法求解[16]。有文章提到利用八个点求 E 得到的解更精确。实际中这些影响可以忽略，因为通常将该结果作为初值，随后通过优化求解。针对单应性矩阵 H（八个自由度），它描述的是两个平面间的运动关系，当特征点都集中在同一个平面上（如无人机俯拍地面），则通过单应性来进行运动估计。H 可以用四组（每三组不共线）匹配特征点采用直接线性变换法（DLT）算出[15]。采用哪种方案求出相机间的运动估计，根据各个不同的应用场合，SVO 采用分解 H 主要用于无人机的俯拍，ORBSLAM 同时求解 E 和 H 进行打分，选择分数高的方案。

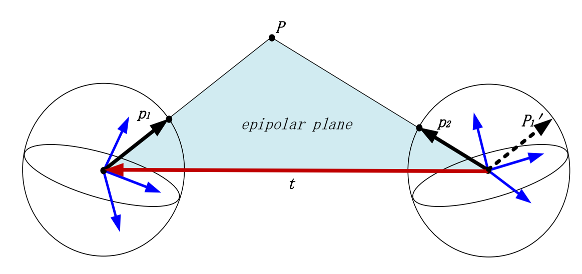


图 6　对极几何视图

Fig. 6　Epipolar geometry view

3D-2D 就是 PnP（perspective-n-point）求解 3D 到 2D 点对运动的方法，描述的是当知道 N 个 3D 空间点及其投影位置时（例如单目，已经初始化完毕，知道特征点的 3D 位置），如何估计相机位姿。当然双目或者深度相机可以直接使用 PnP。对它的求解有 DLT、P3P[17]、EPnP[18]、UPnP[19]。现在常用的做法是先采用 P3P 得到初始解，然后构建重投影误差，使之最小化，如图 7 所示，P_1 和 P_2 是空间点 $P=[X, Y, Z]^T$ 的投影，在初始解中 P 的投影为 $P_2^{'}$，$P_2^{'} = [u_2 \quad v_2]^T = \frac{1}{Z_2} K T_k P = \frac{1}{Z_2} K \exp(\xi^{\wedge}) P$，K 为相机内参，R，t 表示相机外参（李代数为 ξ），Z_2 表示深度值，式中隐含了齐次和非齐次间的转换。如果考虑图像中所有匹配的特征点则得到如下函数：

$$T_k = \begin{bmatrix} R_{k,k-1} & t_{k,k-1} \\ 1 & 1 \end{bmatrix} = \arg\min_{T_k} \sum_i \| p_k^i - p_{k-1}^i \|^2 = \arg\min_{\xi} \sum_i \left\| u_i - \frac{1}{Z_i} K \exp(\xi^{\wedge}) p_i \right\|^2$$

然后使用李代数上的扰动模型分析其导数，并通过高斯牛顿等优化方法得到两帧间的相对变换，具体做法又叫作捆集优化（bundle adjustment，BA）[20]，在编程上一般采用 General Graph Optimization（G2O）等优化库实现。

3D-3D 主要是激光 SLAM 采用迭代最近点（ICP）求解。在 VSLAM 中，可以在 RGB-DSLAM 中使用，但由于 RGB-D 相机的限制，仅仅适用室内，而且适用小的场景。这是由于深度的估计不准，导致误差比 3D-2D 大。直观的感觉是，相机得到的 3D 位置误差较大（相机方向性好，距离信息误差大），3D-2D 只使用一次深度信息，但是 3D-3D 采用两次深度信息，导致计算的精确度降低，所以在普通相机中一般回避 3D-3D 的方式。

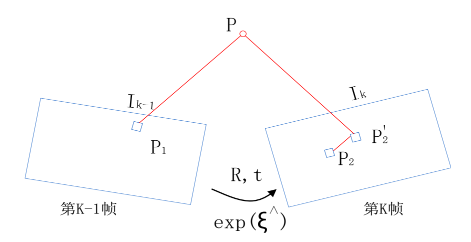


图 7　重投影误差示意图

Fig. 7　Reprojection error diagram

2.1.2 直接法

特征点法有几个问题：

a) 关键点的提取和描述子的计算非常耗时，如果保证 SLAM 实时运行，需要 30 Frame/s，也就是每帧图像的处理时间约 30 ms，而实时性最好的 ORB 也需要近 20 ms/Frame[5]；

b) 特征点法仅仅使用了图像中几百个特征点，占整个图像几十万个像素的很小部分，丢弃了大量可以利用的图像信息；

c) 特征点的寻找是根据人类自己设计的检测算法，并不完善，有些图像没有明显的纹理，有些图像的纹理比较相似，这种情况下特征点法的 VSLAM 就很难运行；

d) 特征点法只能得到空间的稀疏三维点云。离稠密地图尚有一定的距离，与用于机器人导航的地图差距就更大了。

直接法根据像素灰度信息估计相机的运动，几乎不用计算关键点和描述子，省去了计算关键点和描述子的时间，可以在特征点缺失但是有图像灰度梯度的场合（当然对于一张白墙，它也无能为力）。相比于特征点法只能构建稀疏点云地图（构建半稠密或稠密需要采取其他技巧），直接法具备构建半稠密和稠密地图的能力。

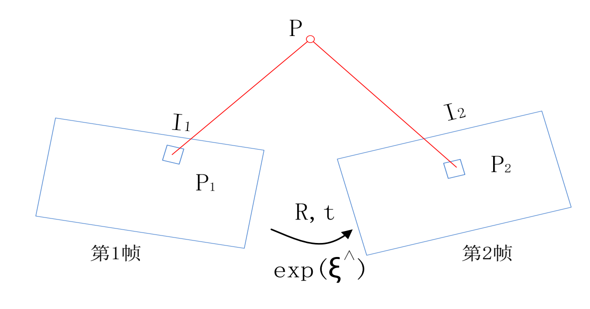


图 8　光度误差示意图

Fig. 8　Photometric error diagram

与特征点法中特征点的特性不变有所不同，直接法的不变量是对应像素点的灰度值。首先假设两个像素点在第一帧与第二帧之间灰度值保持不变，如图 8 所示，P_1 和 P_2 的灰度值是一样的，直接法的思路是根据当前相机的位姿估计来寻找 P_2 的位置，如果相机位姿不好，P_2 和 P_1 的外观会有明显差别。为了减少这个差别，本文优化相机位姿，寻找与 P_1 更相似的 P_2。这就是在灰度不变假设下，直接采用两帧图像中的匹配像素的灰度值，构建光度误差的优化函数，改变相机位姿使之最小化。根据图像像素 P 的情况，直接法分为稀疏、半稠密和稠密直接法。P 如果是稀疏关键点，称之为稀疏直接法；P 如果是图像中梯度明显的点，称之为半稠密法；P 如果是图像中的所有像素，称之为稠密法。

直接法的优化问题构建：考虑某个空间点 $P[X，Y，Z]^T$，P_1、P_2 分别为投影坐标，本文设第一个相机为初始点，第二个相机相对变换为 R、t（李代数为 ξ），Z_1 和 Z_2 是对应的深度值，K 为相机的内参，那么投影方程分别为 $P_1=[u_1 \quad v_1]^T=\frac{1}{Z_1}KP$ 和 $P_2=[u_2 \quad v_2]^T=\frac{1}{Z_2}K(\mathrm{R}P+t)=\frac{1}{Z_2}K\exp(\xi^\wedge)\mathrm{P}$，测量误差为 P_1 和 P_2 的灰度差：$e=I_1(p_1)-I_2(p_2)$。本文的目标是改变相机位姿使所有误差和减小，考虑图像所有像素，构建优化函数（整幅图像像素 P_i 的误差二范数和，优化变量为相机位姿）：$\min J(\xi)=\sum_{i=1}^{N}e_i^Te_i$。其中：$e_i$ 表示图像中所有对应的 P_1 和 P_2 的灰度差。与特征点法一样，也需要推导李代数的导数[5]和采用优化库求解。

同样地，直接法也有自己的局限，首先它需要满足光度不变性假设，这对相机提出了很高的要求，而且稠密法因为需要计算图像的所有像素（640*480 就是 30 万个像素），很难在现有 CPU 上实时运行。在前端，特征点法和直接法最大的区别在于，直接法是依赖于梯度搜索，如果两帧采集时间过大，可能图像运动距离过大，导致灰度不规则变化，从而梯度搜索的优化函数进入局部最小，无法给出较好的优化解；而特征点法对运动和光照有一定的鲁棒性，是根据特征点对距离和光照的鲁棒性来决定的，这也是未来 SLAM 发展的决定因素之一。

2.2 回环检测

回环检测就是利用传感器有效地检测出以前经过这里，它对于 SLAM 系统意义非常重要[22]，因为无论你的数据多么的精确，模型多么的优秀，系统的累积误差始终存在。如果能正确地检测到回环，对构建全局一致性地图是非常有帮助的；从另一方面，可以利用回环检测对跟踪失败后的情况进行重定位。

在 VLSAM 中回环检测大多数做法是基于外观，比较图像间的相似性[23]。如果用特征点的方式，比如采用 SIFT 特征描述一幅图像，首先每个 SIFT 矢量都是 128 维的，假设每幅图像通常都包含 1 000 个 SIFT 特征，在进行图像相似度计算时，这个计算量非常大，所以通常不会直接采用特征点，而是采用词袋模型。

词袋模型(bags of words，BoW)[24]早期是一种文本表征方法，后引入到计算机视觉领域，逐渐成为一种很有效的图像特征建模方法[25]，它通过提取图像特征，再将特征进行分类构建视觉字典，然后采用视觉字典中的单词集合可以表征任一幅图像。换句话说，通过 BoW 可以把一张图片表示成一个向量。这对判断图像间的关联很有帮助，所以目前比较流行的回环解决方案都是采用的 BoW 及其基础上衍生的算法 IAB-MAP[26]。FAB-MAP[27,28]是在滤波框架下计算回环概率，而 RTAB-MAP[29]采用关键帧比较相似性，DLoopDetector[30]（在 DBoW2 基础上开发的回环检测库）采用连续帧的相似性检测判断是否存在回环。

回环检测主要由 BoW 模块、算法模块、验证模块三个部分组成。

BoW 模块分为图像预处理、特征提取、特征聚类和统计得到图像的码本。

a) 图像预处理：假设训练集有 M 幅图像，将图像标准化为 patch，统一格式和规格。

b) 特征提取：假设 M 幅图像，对每一幅图像提取特征，共提取出 N 个 SIFT 特征。

c) 特征聚类：采用 K-Means 算法把 N 个对象分为 K 个簇（视觉单词表），使簇内具有较高的相似度，而簇间相似度较低。

d) 统计得到图像的码本：每幅图像以单词表为规范对该幅图像的每一个 SIFT 特征点计算它与单词表中每个单词的距离，最近的加 1，便得到该幅图像的码本。还需要码本矢量归一化，因为每一幅图像的 SIFT 特征个数不定，所以需要归一化。

算法模块分为贝叶斯估计方法和相似性方法。

贝叶斯估计方法：采用 BoW 描述机器人每一位置的场景图像，估计已获取图像与对应位置的先验概率，对当前时刻，计算该新场景图像与已访问位置匹配的后验概率，概率大于阈值则标记为闭环。

相似性方法：有了字典以后，给定任意特征点 f_i，只要在字典树中逐层查找，最后都能找到与之对应的单词 w_i。通常字典足够大，本文可以说这俩来自同一类物体。但是这种方法对所有单词都是同样对待，常规的做法是采用 TF-IDE（term frequency-inverse document frequency）[31]。TF（某个特征在一副图像中出现的频率）的思想是：某单词在一副图像中经常出现，它的区分度就越高；IDE 的思想是：某单词在字典中出现的频率越低，则图像分类时的区分度越高。设所有特征数量为 n，某个节点 w_i 所含的特征数量为 n_i，那么该单词的 IDF 为 $IDF_i=\log\frac{n}{n_i}$；设图像 A 中单词 w_i 出现了 n_i 次，而一共出现的单词次数是 n，那么 TF 为 $TF_i=\frac{n_i}{n}$；定义 w_i 的权重为 $\eta_i=TF_i\times IDF_i$。将权重应用于图像 A，得到词袋向量 $v_A\triangleq\{(w_1,\eta_1),(w_2,\eta_2),\cdots,(w_N,\eta_N)\}$。通过 L1 范数计算 A、B 图像的相似度 $s(v_A-v_B)=2\sum_{i=1}^{N}|v_{Ai}|+|v_{Bi}|-|v_{Ai}-v_{Bi}|$ [32]。得到相似度评分之后，由于环境千差万别，有的环境外观或十分相似或很大差异，所以采用绝对的相似度阈值很难处理，可以采用先验相似度再归一化或者相对的度量方式。

验证模块主要有时间一致性和结构一致性校验。

时间一致性：正确的回环往往存在时间上的连续性，所以如果之后一段时间内能用同样的方法找到回环，则认为当前回环是正确的，也叫做顺序一致性约束。

结构一致性校验：对回环检测到的两帧进行特征匹配并估计相机运动，因为各个特征点在空间中的位置是唯一不变的，与之前的估计误差比较大小。

目前还没有专门针对直接法的回环检测方法，主流的回环检测都是利用特征点采取 BOW 方式。换句话说回环检测还是依赖于特征点，从这个角度来看，特征点法有很大的优势：特征点法已经提取了特征，直接用这些特征去做回环检测；而直接法没有提取特征，如果想做回环检测，必须要另

外提取特征。这也是 ORBSLAM 和 LSDSLAM 中的回环检测采取的不同的方式。

ORBSLAM 中的回环检测与整个系统结合得比较紧密，整个系统都是采用的 ORB 特征，首先离线训练得到 ORB 词典，在搜索时因为 ORBSLAM 本身就已经计算了特征点和描述，可以直接用特征来搜索，而且 ORBSLAM 采用正向和反向两种辅助指标：反向指标在节点（单词）上储存到达这个节点的图像特征的权重信息和图像编号，因此可用于快速寻找相似图像。正向指标则储存每幅图像上的特征以及其对应的节点在词典树上的某一层父节点的位置，因此可用于快速特征点匹配（只需要匹配该父节点下面的单词）。

LSDSLAM 是采用 OpenFABMAP（OpenCV 上实现的 FAB-MAP）来完成回环功能。FAB-MAP 在贝叶斯框架下，采用 Chou-Liu tree[33]估计单词的概率分布，能够完成大规模环境下的闭环检测问题，但是它通过连续的当前帧数据与历史帧数据比较，效率较低，不能满足实时地回环检测。个人感觉 LSDSLAM 中的回环检测是为了完成这个大的系统，额外添加的模块，其实与系统契合度不是很高。

3　后端

3.1　后端优化

SLAM 的后端求解方法可大致分为两大类，一类是基于滤波器的方法；另一类则是非线性优化方法。这是根据假设的不同，如果假设马尔可夫性，K 时刻状态只与 K-1 时刻状态有关，而与之前的状态无关，这样会得到以扩展卡尔曼滤波（EKF）为代表的滤波器方法。在滤波方法中，本文会从某时刻的状态估计推导到下一个时刻。另外一种方法是考虑 K 时刻与之前所有状态的关系，这将得到非线性优化为主体的优化框架[5]。

3.1.1 滤波方法

最早定位和建图是作为两个独立的领域进行研究，在文献[34]中证实可以统一到一个框架中保持收敛；然后由于 SLAM 本质上是一个状态估计问题，该问题可以归结为一个运动方程和一个观测方程，顺理成章地把 SLAM 融入到滤波框架中。早期的 SLAM 研究基本都是在滤波器的框架下。在假定从 0 到 t 时刻的观测信息以及控制信息已知的条件下，对系统状态的后验概率进行估计，根据后验概率表示方式的不同，存在多种基于滤波器的方法，如扩展卡尔曼滤波（EKF）方法、粒子滤波（PF）等。

第一个实时单目 VSLAM 是帝国理工大学的 Davison 等人[35]在 2006 年发布的 MonoSLAM。它以扩展卡尔曼滤波为后端，追踪前端非常稀疏的特征点，以相机的当前状态和所有路标点为状态量，更新其均值和协方差。图 9 所示是 MonoSLAM 在运行时的情形。可以看到，单目相机在一幅图像当中追踪了一些稀疏的特征点，所以本文能够以一个椭球的形式表达它的均值和不确定性。在该图的右半部分，本文可以找到一些在空间中分布着的小球。它们在某个方向上显得越长，说明在该方向的位置就越不确定。可以想象，如果一个特征点收敛，应该能看到它从一个很长的椭球（相机 Z 方向上不确定性很大）最后变成一个小点的样子（在 EKF 中，假设每个特征点的位置服从高斯分布，如果一个特征点收敛，那它最后汇聚为一个小点）。

该工作在当时已经是里程碑的工作了，因为在此之前的视觉 SLAM 系统基本不能在线运行，只能事先使用相机采集数据，然后离线地进行定位与建图。2012 年 Kim 在原版的基础上实现了加强，加入了 Eigen 和 Panglion 库，而且可以使用 USB 相机（早期的版本只能使用网口相机）。但是该框架存在应用场景窄、路标数量有限等限制，仅仅用于实验室内小规模环境下的相机姿态定位和环境构建，后面对它的开发也已经停止。

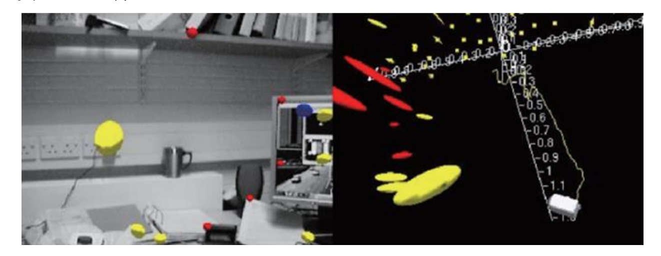


图 9　MonoSLAM 的运行显示图

Fig. 9　Monoslam operation display

随着 SLAM 问题研究的深入及其应用逐步从小场景转向大场景，基于滤波的 SLAM 方法越来越受到局限。比如 EKF 方法需要把路标放进状态，由于 VSLAM 中路标数量很大，而且储存的状态量呈平方增长（协方差矩阵），所以 EKFSLAM 被普遍认为不适合大型场景。再者，滤波方法假设马尔可夫性，假设当前状态只与上一时刻相关，而与之前状态和观测都无关，这种处理方式使得滤波器很难处理回环等问题。

而基于非线性优化方法倾向于使用所有的历史数据，称为全体 SLAM（full SLAM）。从某种程度上说，非线性优化使用了更多的信息，当然能获得更好的建图效果。Strasdat 等人[36]证明了在相同的计算单元下，基于优化的方法比基于滤波的方法能够获得更高的精度。

3.1.2 非线性优化方法

代价函数的建立：在 VSLAM 中，如果不考虑运动方程，假设观测误差：$e=z-h(\xi,p)$，其中 h（）为观测方程，ξ 为外参 R，t 对应的李代数，三维点 P 是路标，k 像素坐标 $z=[u_k,v_k]^T$。如果考虑所有的观测量，那么整体的代价函数为 $\sum_{i=1}^{m}\sum_{j=1}^{n}\|e_{ij}\|^2=\sum_{i=1}^{m}\sum_{j=1}^{n}\|z_{ij}-h(\xi_i,p_j)\|^2$，对这个函数采用最小二乘求解，相当于对所有相机位姿和路标同时调整，使目标函数最小，这就是 bundle adjustment （BA）[19]。过去，研究者普遍认为非线性优化方法计算量非常大，不适合实时计算；直到最近十年，SLAM 问题中 BA 的稀疏特性才逐渐被认识到，才使它能够在实时的场景中应用[37]。

对上式 BA 的求解，无论是采用高斯牛顿还是列文伯格—马夸尔特（LM）方法，最后都将面临增量方程：$H\Delta x=g$。以高斯牛顿为例，H 矩阵为 $H=J^TJ$，由于雅可比矩阵 J 包含了所有的路标点，尤其是 VSLAM 中，一幅图像至少会提取数百个特征点，如果直接对 H 求逆（复杂度为 O（n3）），计算量非常大。

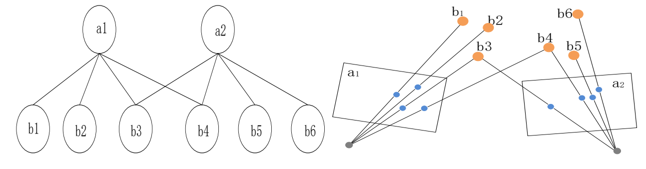


图 10　观测示意图

Fig. 10　Observation diagram

矩阵 H 的稀疏结构[21]：假设场景中有两个相机位姿（a1，a2）和六个路标（b1，…，b6）（图 10），a1 观测到路标 b1、b2、b3、b4，a2 观测到路标 b3、b4、b5、b6，则雅可比 J 为 8*8 的矩阵（两个相机位姿加六个路标），具体表示如下所示：

如图 11 所示，$A_{ij}=\partial\hat{X}_{i,j}/\partial a_i$ 是关于相机位姿的雅可比，表示由于相机位姿 a_i 的改变而引起 $\hat{X}_{i,j}$ 的改变；同理 $B_{ij}=\partial\hat{X}_{i,j}/\partial b_j$ 是关于 3D 点的雅可比，表示由于 3D 点 b_j 的改变而引起 $\hat{X}_{i,j}$ 的改变，其中 $\partial\hat{X}_{i,j}/\partial a_k=0$ 当 $j\neq k$，因为改变相机位姿 a_k 不影响因为 a_i 引起的估计 $\hat{X}_{i,j}$；同理 $\partial\hat{X}_{i,j}/\partial b_k=0$ 当 $j\neq k$，因为改变 3D 点 b_k 不影响因为 b_j 引起的估计 $\hat{X}_{i,j}$，也就是说当 $j\neq k$ 时，它们是无关的。比如考虑其中一个 e_{ij}，它只描述了在 a_i 看到 b_j 这件事，只涉及第 i 个相机位姿和第 j 个路标点，对其余部分的变量的导数都为 0。更简单地说，残差 e_{11} 表示在 a_1 看到了 b_1，与其他的相机位姿和路标无关，J_{11} 为 e_{11} 所对应的雅可比矩阵，从而得到如图 12（a）所示 的雅可比形式，再进一步得到如图 12（b）所示 的矩阵 H（H 的稀疏性是由 J 引起的）。因为 VSLAM 中路标数量至少也有数百个，所以矩阵 H 的右下角是一个维数很大的对角块矩阵，该对角块求逆难度远小于对矩阵 H 的求逆难度。鉴于此，对 BA 的求解都是采用 Schur 消元（也称做边缘化），具体 BA 的算法一般采用 G2O[38]或 Ceres[39]库实现。

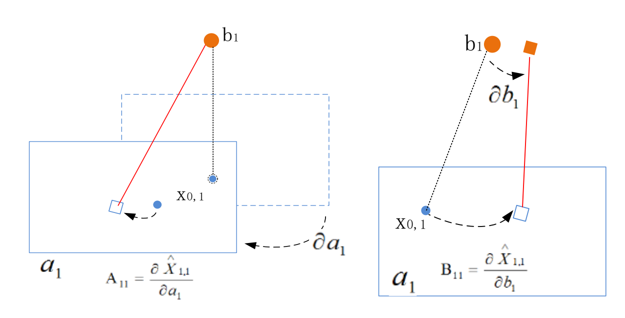


图 11　Aij 和 Bij 示意图

Fig. 11　Aij and Bij schematic

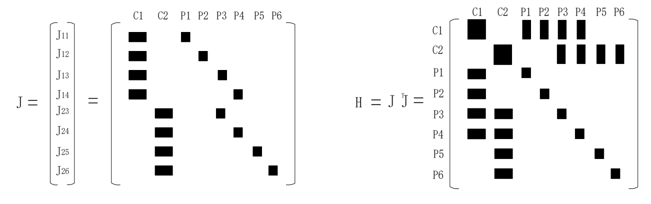


图 12　J 矩阵和 H 矩阵的稀疏结构示意图

Fig. 12　Sparse Structural Diagrams of J Matrix and H Matrix

随着计算机性能的进步，以及逐渐认识到 VSLAM 中雅可比矩阵的稀疏特性，现在主流的 VSLAM 都是采用非线性优化的方法，使用 G2O 等库来求解 BA。VSLAM 的后端仅仅出现过一个基于滤波器的 MonoSLAM，之后都是非线性优化统一了后端：一是 EKF 需要对地图和相机位置进行更新，但是 VSLAM 中路标的数量动辄成百上千，存储的状态量呈平方增长，所以 EKF 被普遍认为不适合大的场景，而优化方法没有这样的限制；还有非线性优化可以利用历史所有数据，这和回环检测的模型是相关的，而 EKF 很难做回环检测。

3.2　建图

地图的具体形式主要有路标地图、拓扑地图、度量地图和混合地图。

路标地图，由一堆路标点组成，在早期的基于 EKF 的 SLAM 中比较常见。

拓扑地图强调地图元素之间的连通关系，由节点和边组成，只考虑节点间的连通性，而对精确的位置要求不高，去掉了大量地图的细节，是一种比较紧凑的地图表达方式。

度量地图分为栅格地图和几何地图。栅格地图将整个环境分为若干个大小相同的栅格，每个栅格代表环境的一部分。二维栅格地图在以激光雷达为传感器的扫地机器人里十分常见，它只需用 0-1 表示某个点是否有障碍，对导航很有用，而且精度也比较高，但是它比较占存储空间，尤其是三维栅格地图，它需要把所有的空间点都存起来。几何地图通过收集对环境的感知信息，从中提取几何特征（如点、线、面）描述环境，多见于早期的 SLAM 算法中。

混合地图通常采用分层结构将多种地图组合，如拓扑地图和度量地图组成的混合地图，上层的拓扑地图实现粗略的全局路径规划，底层的度量地图实现精确的定位和路径的优化。

在 VSLAM 中广泛应用的是度量地图，它精确地表示地图中物体的位置关系，可按稀疏和稠密划分。特征点法得到稀疏点云地图，直接法得到半稠密或稠密地图。针对稠密的度量地图，当查询某个空间位置时，地图能够给出该位置是否可以通过的信息。

VSLAM 中建图的基本原理是通过三角测量或深度估计，将 2D 图像中的信息转换为空间 3D 路标点。在 VSLAM 中建图过程和位姿估计过程是同时完成的。

在单目 VSLAM 中，仅仅通过单张图像无法获得像素 3D 信息，需要通过三角测量来进行估计。一方面，由于噪声存在，无法得到精确解；另一方面，当平移很小时，像素上的不确定性将导致较大的测量不确定性，平移较大时，在相同的相机分辨率下，三角测量将更精确。它有如下矛盾：平移增大，会导致匹配失效；平移太小，三角化精度不够。因此可通过多帧图像来减少 3D 点的不确定度或采用尽可能宽的极线来获得 3D 信息。

深度估计在建图模块中占据非常重要的地位，通常在 VSLAM 系统中都有专门的线程对其进行处理。SVO 采用高斯加上均匀分布的方法估计三维空间点的深度信息，并不断更新，直到其收敛。在 LSDSLAM 中，针对关键帧，通过之前关键帧的点投影初始化当前帧的深度估计；针对非关键帧，通过卡尔曼滤波不断地利用观测值对深度进行修正。

4　开源算法比较

按照特征法和直接法的分类，各种 VSLAM 具备不同的处理速度、轨迹精度等指标，如表 1 所示。随着 VSLAM 的研究如火如荼地开展，许多研究者发表研究成果以及公开相关代码，供学者学习与研究。下面针对 VSLAM 发展历程中几个最具代表性的开源系统进行详细介绍与综述。

表 1　VSLAM 分类比较

Table 1　VSLAM classification comparison

	特征法	直接法	混合法
处理速度	☆	☆☆	☆☆☆
估计轨迹精度	☆☆☆	☆☆	☆
适应场景能力	☆	☆☆☆	☆☆
硬件适应性	☆☆☆	☆	☆☆
初始化适应性	☆☆☆	☆	☆☆
构建地图能力	☆☆	☆☆☆	☆
可扩展性	☆☆☆	☆	☆
信息利用率	☆	☆☆☆	☆

4.1 特征点法

4.1.1 PTAM

PTAM[40]是 2007 年由牛津大学主动视觉实验室的 Georg Klein 和 David Murray 提出的。当时给研究者们带来了极大震撼，它有如下创新点：

a) PTAM 第一个使用非线性优化。之前人们未认识到后端优化的稀疏性，所以觉得优化后端无法实时处理那样大规模的数据，主流的 SLAM 均采用 EKF 滤波器等滤波方法。而 PTAM 则是一个显著的反例，将 VSLAM 研究逐渐转向了以非线性优化为主导的后端。

b) PTAM 引入了关键帧机制。不必精细地处理每一幅图像，而仅仅处理较少的关键帧图像，然后优化其轨迹和地图。

c) PTAM 引入了多线程机制。将跟踪和建图过程分开。因为跟踪部分需要实时响应图像数据，而地图则没必要实时地优化，只需在后台进行处理。这是 VSLAM 中首次区分出前后端的概念，初步确定了 VSLAM 的框架。

PTAM 主要分为跟踪、建图两部分。

PTAM 的跟踪分为粗阶段和精阶段。在粗阶段中选用图像金字塔最高层的 50 个特征点，利用恒速模型和扩大范围搜索，从这些测量中得出一个新姿态；再将近千个特征点重新投影到图像中，执行更严格的块搜索（FAST 特征的局部 8*8 的方块构成 patch 作为描述符），并构建重投影误差得到最优的相机姿态。

地图构建主要是建立三维地图点的过程。它分为地图的初始化和地图的更新。首先，系统初始化时使用三角测量构建初始地图；在此之后，随着添加新的关键帧，地图将不断地进行细化和扩展。具体为：系统初始化时，根据前两个关键帧提供的特征对应关系，采用 5 点算法和随机采样一致（RANSAC）估计本质矩阵（或使用平面情况的单应性分解）并三角化得到初始地图。然后当插入关键帧时，使用极线搜索和块匹配（零均值距离平方和 ZMSSD）计算得到精确匹配，从而精细化地图。PTAM 系统框图如图 13 所示。

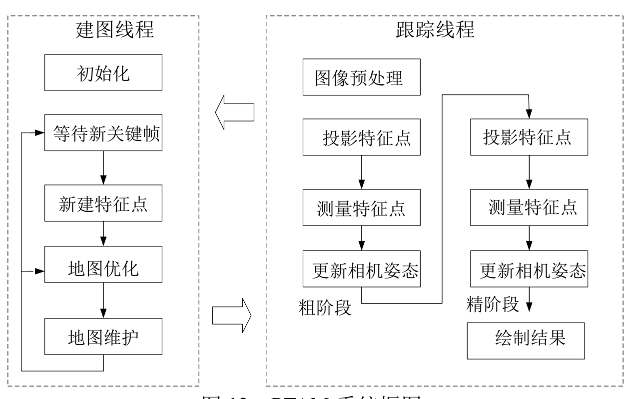


图 13　PTAM 系统框图

Fig. 13　PTAM system block diagram

PTAM 不仅仅是 VSLAM 的程序，还将相机的标定和增强现实（AR）都包括进来，而且试图在手机上实现，从另外的角度也可以说它是面向小场景的一个增强现实软件。PTAM 的最开始的版本是建议采用 5 点算法[41]分解本质矩阵得到相机姿态，该方法用于非平面场景的初始化。后来 PTAM 的初始化改变为使用单应性[42]，其中场景假定为 2D 平面。以现在的知识来看，PTAM 的 DEMO 可能有点过时，比如它的初始化需要用户的输入来捕捉地图中的前两个关键帧，而且它要求用户在第一与第二关键帧之间采取平行于观察场景的缓慢和平滑的平移运动。因为它采用的 2D-2D 的图像匹配算法为不考虑特征仿射变换的 ZMSSD 算法，所以容易受到运动模糊和相机旋转的影响。

PTAM 是为小场景 AR 设计的，没考虑全局的回环，而且存在明显的缺陷：场景小（实际情况是 6 000 个点和 150 个关键帧）、跟踪容易丢失等，但是在当时确实是一个里程碑的标志。

4.1.2 ORBSLAM

ORBSLAM[43,44]由西班牙 Raul 博士于 2015 年公布，其论文发表在《IEEE Transactions on Robotics》。到目前为止，ORBSLAM 是最完整的基于特征点法 VSLAM，它可以看做是 PTAM 的一个延伸，相比 PTAM，ORBSLAM 增加了一个回环检测（loop closing）的线程。该系统框架包括跟踪、建图、闭环三个线程，均基于 ORB 特征实现，所有优化环节都通过优化框架 G2O 实现。

ORBSLAM 有如下创新点：

a) 初始化采用自动机制，不需要手工输入，也不需要假设场景是否为平面。通过匹配 ORB 特征同时计算单应性和基础矩阵并评分，选用分数高的方案。

b) 将改进后的 ORB 特征贯穿整个工程始终，包括特征检测、匹配以及用于闭环的词袋模型（bag-of-words，BoW）[45]。

c) 使用 DBOW 模块，不只是用于 loop closing 时的检测，而且用于系统的重定位。更大的意义是在图像帧间匹配时，使用词典对描述子进行分类的结果进行比对，这种方法不仅有效，还可以大大简化运算。

d) 后端优化是亮点，ORB 在每一层估计中都大量采用 G2O 优化，不仅有单帧位姿估计到局部地图的位姿估计，而且有局部地图点与位姿联合估计，还有利用回环结果的全局位姿估计。

ORBSLAM 的具体流程为：

a)跟踪。跟踪线程主要是得到相机位姿和关键帧。具体为：先对图像进行 ORB 特征提取和匹配，系统初始化得到 R，t 和 3D 点云（如果系统未初始化）；然后采用参考关键帧模型或运动模型和 BoW 模块加速匹配（如果跟踪失败也是将当前帧和所有关键帧通过 BoW 加速匹配），再构建局部小图和重投影误差优化函数；最后得到优化位姿和关键帧。

b)建图。跟踪线程主要是更新 3D 点和插入关键帧。具体为：取出一个关键帧，计算特征点的 BoW 关系，更新关键帧间的连接关系，将关键帧插入地图，验证加入的地图点，利用三角法生成新的地图点，对相邻关键帧和对应的 3D 点进行局部 BA，剔除冗余关键帧，将关键帧加入闭环；

c)闭环。闭环线程主要是纠正尺度漂移和全局优化。具体为：取出一个关键帧，计算当前关键帧与每个共视关键帧的 BoW 得分，在所有关键帧中找出闭环备选帧，通过连续性检测验证候选帧，去做 sim3 优化[46]（纠正尺度漂移，使其尺度一致），利用优化结果寻找更多的特征匹配，再做一遍优化，如果内点足够，接收这个闭环，最后固定回环帧和当前帧再做全局优化。

ORBSLAM 在工程上是非常完整的 SLAM 系统，里面涉及的很多参数都是通过计算得出，后续有大量的学者在其基础上改进。随后 Raul 在前面的基础上利用宽基线做了更加精密的半稠密地图构建的工作[47]，2017 年又将 IMU 融入到 ORBSLAM 中[48]，由此可见 ORBSLAM 的可扩展性很好。当然与许多其他的基于特征点的 SLAM 系统一样，有很多自身的缺陷：因为特征点的原因，只能得到稀疏点云地图，这对机器人下一步的导航应用会造成很大困难，而且它不易作为环境地图的描述，也很难构建高层次地图（语义地图等），给

环境的语义构建带来了诸多不便。

特征点法中的特征一般为本文人类根据图像的一些特性，自己设计的算法，可能失去了大自然的本质和意义；在特征点法中绝大多数时间都耗费在特征的提取和匹配上，特征点法的瓶颈在于如何设计和提取更好的特征点；相比之下，直接法不依赖特征的提取和匹配，直接通过两帧之间的像素灰度值构建光度误差来求解相机运动。因此直接法可以在特征缺失的场合下使用。

4.2 直接法

4.2.1 LSDSLAM

DTAM[49]是直接法的鼻祖，是 2011 年提出的单目 SLAM 算法，对每个像素点进行概率的深度测量，有效降低了位姿的不确定性。该方法通过整幅图像的对准来获得稠密地图和相机位姿，但是需要 GPU 加速，超出了本文的讨论范围。

基于同样的原理，TUM 机器视觉组的 Engel 等人[50]于 2013 年提出了基于直接跟踪的视觉里程计(semi-dense visual odometry)系统，该 VO 系统是第一个不采用特征的实时的视觉里程计。后来他将地图优化融入该 VO 系统并扩展为 LSD-SLAM[51]，得到了不采用特征的实时的 SLAM 系统。该系统通过对图像光度直接配准和使用概率模型来表示半稠密深度图，生成具有全局一致性的地图。

它具有如下创新点：

a) 使用随机深度初始化策略类似于滤波器方法的思路来完成初始化。将图像中的像素以随机的深度初始化，并利用新产生的数据不断迭代优化直至收敛，当初始场景的深度方差收敛到最小值时，认为初始化完成。

b) 通过假设图像像素逆深度服从高斯分布，对每个像素深度独立计算，通过卡尔曼滤波更新深度估计，将深度图的噪声融合到图像跟踪中，构建半稠密和高精度的三维环境地图；

c) 为了避免尺度上的漂移，将估计的深度均值归一化，而且考虑深度和极线的夹角，在关键帧的直接配准上，采用 sim3 来衡量其变换，并将光度残差和深度残差一起放入优化函数中：$E(\xi_{ji}) := \sum_{p\in\Omega_{Di}} \left\| \frac{r_p^2(p,\xi_{ji})}{\sigma_{r_p}^2(p,\xi_{ji})} + \frac{r_d^2(p,\xi_{ji})}{\sigma_{r_d}^2(p,\xi_{ji})} \right\|_\delta$。其中：两项分别为被归一化的光度残差和深度残差；$\| \|_\delta$ 表示 Huber 核函数，避免误差太大而覆盖其他的正确值。

LSDSLAM 具有图像跟踪、深度估计和地图优化三个线程。

a)图像跟踪。主要计算当前帧与参考帧之间的相对变换，有精确方式和快速方式，都采用加权的高斯牛顿优化方法。关于跟踪失败后的重定位，在本文里没有，但是开源代码里有实现：将当前帧和邻近的关键帧连接起来，计算坐标变换关系，通过打分和遍历整个附近帧来判断是否完成重定位。

b)深度估计。当相机移动超过了阈值，那么需要创建关键帧，将之前关键帧的点投影到当前新的关键帧上，通 sim3 变换得到该关键帧的深度估计。当跟踪帧没有变为关键帧，本文用它来更新图像的像素深度：先采用自适应的方法确定搜索范围，然后通过卡尔曼滤波不断地利用观测值对深度进行修正。

c)地图优化。其目的是利用闭环解决尺度漂移的问题。首先去寻找所有可能相似的关键帧，并计算视觉意义上的相似度，并由 appearance-based mapping 算法[52]筛选出来的候选帧，而且需要进行跟踪检测，当完成闭环约束以后，再通过全局优化得到全局一致性地图，其包括关键帧组成的姿态图和对应的半稠密深度图。

LSDSLAM 是直接法中比较完整的 SLAM 系统，能够在普通 CPU 上实现半稠密 SLAM（梯度明显的像素），后续 Engel 对 LSDSLAM 进行了功能拓展，使其能够支持双目相机[53]和全景相机[54]。但是它存在一定缺点：对相机内参和曝光非常敏感，而且准确性方面不及 ORBSLAM，速度方面不及 DSO，作者后续研究了光度标定，将其扩展应用于 DSO 系统（https://github.com/JakobEngel/dso）。

4.2.2 DSO

DSO[55]为 Engel 在 2016 年发布的一个视觉里程计方法，因为没有闭环，所以只能算 SLAM 的一个模块（后续应该会完善），文中宣称速度可以达到传统特征点法的五倍。

直接法因为是比较两帧图像之间的像素差异，需要满足光度不变，但是这是一个很强的假设，尤其是针对普通的自动曝光相机。在做 DSO 工作之前，Engel 先研究了光度标定相关工作，因为他认为对相机的曝光时间、暗角、伽马响应等参数进行标定后，能够让直接法更加鲁棒[56]。这个过程建模了相机的成像过程，对于由相机曝光不同所引起的图像明暗变化会有更好的表现。DSO 是一种结合直接法和稀疏法的视觉里程计，它不检测和计算特征点，而是采样图像内具有强度梯度的像素点；它将光度误差模型和所有模型参数融入到优化函数中进行联合优化，而且该系统结合曝光时间、透镜晕影以及非线性响应函数的影响提出了完整的光度标定方法，并在多个数据集上进行了测试，达到了很好的精度和速度，可以说是 LSDSLAM 的升级版。进一步地，Engel 小组研究了双目的 DSO，但并没有开源代码（吴佳田、颜沁睿等做了相应的工作（https://github.com/HorizonAD/stereo_dso）），而且包括他们自己在内的很多研究者在 DSO 的基础上扩展，尝试给 DSO 添加回环检测和地图重用的模块。

将相机内参和曝光参数作为优化变量引入优化函数，并推导了其相对于残差的雅可比是 DSO 的最大创新之处。其流程如图 14 所示。首先是两帧图像对齐初始化和地图点的更新：地图点在一开始被观测到时，其深度是未知的，随着相机的运动，DSO 会采用沿着极线搜索方式在每张图像上追踪这些地图点，跟踪过程会确定每个地图点的逆深度和变化范围；然后通过相机视野改变、相机平移和曝光时间显著改变这些参数是否达到阈值来构建关键帧。在后端优化过程中，DSO 采用由相个关键帧组成滑动窗口的方式，不断地计算需删除的关键帧和添加关键帧，并且将每个先前关键帧中的地图点投影到新关键帧中，形成残差项，同时在新的关键帧中更新地图点和删除外点。

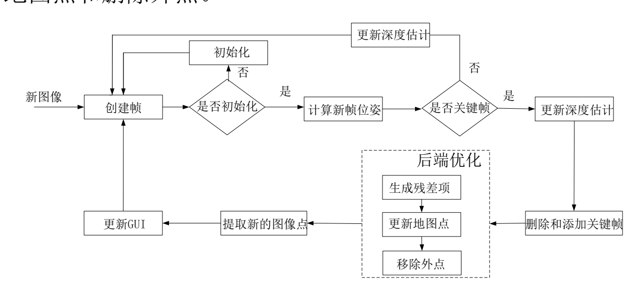


图 14 DSO 的系统流程

Fig. 14 DSO system flow chart

光度误差：$E_{pj} := \sum_{p\in N_p} w_p \left\| (I_j[p']-b_j) - \frac{t_j e^{a_j}}{t_i e^{a_i}} (I_i[p]-b_i) \right\|_r$，其中：$N_p$ 表示投影点和周围的点组成一个包含八个点的图案（pattern），右下角为空，在 DSO 中，本文假设这八个点在不同图像中保持灰度不变；w_p 表示梯度加权，梯度越高权重

越低；p' 为 P 点在当前图像 j 中的投影位置；t_i、t_j 分别为图像 i、j 的曝光时间；a_i、a_j、b_i、b_j 为亮度传递函数的参数。

直接法是采用大量的像素信息来优化求解相机位姿，与特征点法相比是数量替代质量的过程，DSO 初始化不仅需要较好的初始估计，还比较依赖梯度下降的优化策略，而它成功的前提要求目标函数是单调的，但这件事情往往无法得到保证。进一步说，如果想在 DSO 上加重定位功能，首先需要保存所有帧，然后需要对相机位姿有一个比较准确的初始估计。但这通常是困难的，因为不知道误差累积了多少。而在特征点法中，地图重用则相对简单。本文只需存储空间中所有的特征点和它们的特征描述，然后匹配当前图像中看到的特征，计算位姿即可。从这个角度来看：直接法应该更擅长求解连续图像的定位，而特征点法则更适合全局匹配与回环检测。

4.3 特征点法和直接法的结合

4.3.1 SVO

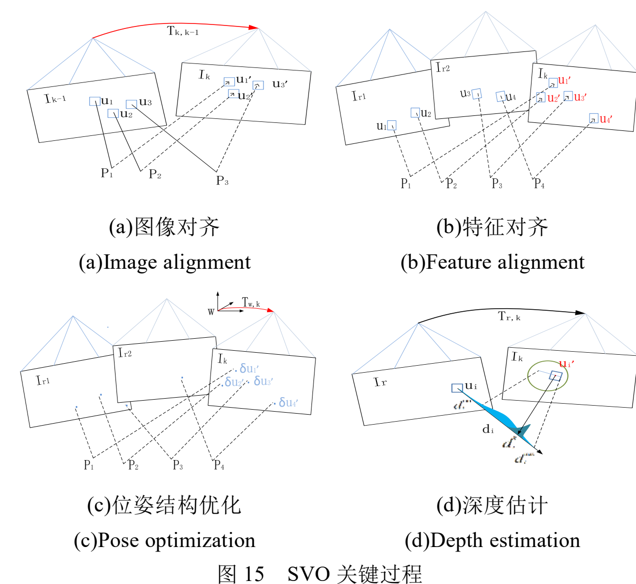
特征点法精度高，直接法速度快，两者是否可以结合呢？苏黎世大学机器人感知组的 Forster 等人[57]2014 年提出的一种半直接法的视觉里程计（SVO），半直接是指通过对图像中的特征点图像块进行直接匹配来获取相机位姿，而不像直接匹配法那样对整个图像使用直接匹配。SVO 面向无人机航拍场合，将特征点法与直接法结合，跟踪关键点，不计算描述子，根据关键点周围的小图像块的信息估计相机的运动。主要分运动估计线程和地图构建线程两个线程，其中运动估计分为如下三步：

a)图像对齐。如图 15（a）所示，通过当前帧和参考帧中的特征点对的 patch(特征点周围 4*4 区域)的灰度差异，构建光度误差的优化函数 $T_{k,k-1}=\arg\min_{T_{k,k-1}}\frac{1}{2}\sum_{i\in R}\|\sigma I(T_{k,k-1},u_i)\|^2$，其中：$\sigma I$ 为像素 u 在图像 K 和 K-1 的灰度差，表示为 $\sigma I(T,u)=I_k(\pi(T\cdot\pi^{-1}(u,d_u)))-I_{k-1}(u)$；$\pi,\pi^{-1}$ 表示投影和反投影；优化变量为相机的变换矩阵 T，采用高斯牛顿迭代求解，然后寻找更多的地图点到当前帧图像的对应关系。

b)特征对齐。如图 15（b），由于深度估计和相机位姿的不准导致预测的特征块位置不准，通过光流跟踪对特征点位置进行优化，具体为：对每个当前帧能观察到的地图点 p（深度已收敛），找到观察 p 角度最小的关键帧 r 上的对应点 u_i，得到 p 在当前帧上的投影。优化的目标函数是特征块（8*8 的 patch）及其在仿射变换下的灰度差 $u_i^{'}=\arg\min_{u_i^{'}}\frac{1}{2}\|I_k(u_i^{'})-A_i\cdot I_r(u_i)\|^2$，其中：$A_i$ 表示仿射变换；优化变量为特征点位置 u_i’。

c)位姿结构优化。如图 15（c）所示，像素位置优化后，利用建立的对应关系对空间三维点和相机位置进行分别优化，构建像素重投影误差的优化函数 $T_{k,w}=\arg\min_{T_{k,w}}\frac{1}{2}\sum_i\|u_i-\pi(T_{k,w}\,{}_wp_i)\|^2$，优化变量为相机的变换矩阵 T 或空间三维点 P。

地图构建主要是深度估计，如图 15（d）所示，它采用文献[58]中的概率模型，高斯分布加上一个设定在最小与最大深度之间的均匀分布：$P(d_i^k|d_i,\rho_i)=\rho_iN(d_i^k|d_i,\tau_i^2)+(1-\rho_i)u(d_i^k|d_i^{\min}{}_i,d_i^{\max})$，并推导了均匀—高斯混合分布的深度滤波器，采用逆深度作为参数化形式。当出现新的关键帧时，选取若干种子点，每个种子点根据变换矩阵得到对应的极线，在极线上找到特征点的对应点，通过三角测量计算深度和不确定性；然后不断更新其估计，直到深度估计收敛到一定程度，将该三维坐标加入地图。



(a)图像对齐　　(b)特征对齐

(a)Image alignment　　(b)Feature alignment

(c)位姿结构优化　　(d)深度估计

(c)Pose optimization　　(d)Depth estimation

图 15　SVO 关键过程

Fig. 15　SVO key process

SVO 在运动估计中的思路很新颖，运行速度非常快，由于不用计算描述子，也不用处理过多的地图点云，在普通 PC 上也能达到 100 Frame /S 以上。但是正因为它的目标应用平台是无人机，所以它在其他的场合应用是不适合的，至少是需要修改的。例如在单目初始化时，是采用分解 H，这需要假设前两个关键帧的特征点位于一个平面上。再者，在关键帧的选取策略上，采用的是平移量，没有考虑旋转。而且它是一个轻量级的相机运动估计，没有闭环功能，没有重定位，建图功能也基本没有，即使如此，它也不失为一个优秀的 SLAM-DEMO。2016 年，Forster 等人[59]对 SVO 进行改进，形成 SVO2.0 版本，新的版本作出了很大的改进，增加了边缘的跟踪，并且考虑了 IMU 的运动先验信息，支持大视场角相机（如鱼眼相机和全景相机）和多相机系统，该系统目前也开源了可执行版本（http://rpg.ifi.uzh.ch/svo2.html）。值得一提的是，Foster 对 VIO 的理论也进行了详细的推导，尤其是关于预积分的论文[60]成为后续 VSLAM 系统融合 IMU 的理论指导。

5　特征点法和直接法的发展方向

一方面，虽然直接法在某种程度上缓解了对特征的依赖，而且可以得到半稠密乃至稠密地图，但所需的计算量很大；另一方面，针对一些人造环境，结构化特征比较丰富，如线特征、面特征等，当然可以基于线面特征考虑 SLAM。

确切地说：特征点法不过是特征法中选用点特征而已。因为点特征研究得最多、最广、最透彻，而线、面的检测以及描述不像点特征那么丰富和成熟。即便如此，基于线、面以及点、线、面结合的 VSLAM，已经有很多学者进行了相关研究。

早期，Smith[61]采用两点表示空间直线的方式实现 VSLAM，因为需要保证相机观测到这两个端点，所以该系统适用于小场景。Sola 等人[62]在此基础上提出将空间直线用无限延长的线段来表示，所以该 VSLAM 适合较大距离的场景。Eade 等人[63]通过跟踪短线段和点特征，在滤波框架下进行位姿估计。随着优化方法的崛起，基于线特征的 VSLAM 研究逐步转移到优化框架中：PL-SVO[64]利用点、线特征的组合描述能力，提出了基于点线结合的双目视觉里程计，在优化

时对点、线特征的重投影误差采取不同的权重。在此基础上，PL-SLAM[65]通过词袋模型实现了回环检测，而且通过点线结合得到的地图更丰富，更容易得到高层次场景结构。Lee 等人[66]首先用 MSLD 线段描述子构建字典树进行场景识别，而且使用线特征实现了实时的闭环检测，随后他们利用线特征在室外场景实现位置识别算法[67]，并在上万张真实世界的图像数据库中测试成功。进一步地，Zhang 等人[68]在此基础上实现了 SLSLAM（stereo line-based SLAM），该系统提出了基于线段特征的 SLAM 框架，包括利用线特征完成运动估计、位姿优化、闭环检测等，构建了目前较为完善的基于线段特征的 SLAM 系统。Zuo 等人[69]针对直线特征采用正交表示法作为最小参数化，而且推导出了基于线特征的误差函数的雅可比矩阵解析形式。

基于面特征的研究有：2011 年 ETH 的 Lee 等人[70]提出通过面约束来减少 BA 的计算量；文献[71]使用深度相机在两个不同坐标系下完成点、面的配准，并在 BA 框架下实现了点面结合的 SLAM 系统；Yang 等人[72]针对低纹理环境提出单目平面 SLAM 方法，并验证该方法能够改善状态估计和地图构建；李海丰等人[73]为了减少点特征的计算量和误差大的问题，在 EKF 框架下提出了基于点、线段、平面特征融合的 VSLAM 算法（PLP-SLAM），并在数据集上进行了验证。

虽然线面特征的 VSLAM 有一定的发展，但是在理论上还需要丰富线面特征的描述、提取和匹配；在应用上它们比特征点法适用范围窄，但是将其作为人造环境中的辅助和高层次表达是可行的。

近来深度学习广泛流行，它的主要优势是在物体识别方面，尤其是计算机视觉领域，主流的识别算法几乎都采用深度学习。而 VSLAM 框架中的视觉里程计和回环检测都是与图像的检测和识别相关联，所以将深度学习用于 VSLAM 中的前端是顺理成章的事情。广义上说，直接法 VSLAM 就是直接通过图像得到相机位姿估计。目前深度学习与 VSLAM 的结合主要是利用深度学习的方法完成视觉里程计模块和回环检测模块，也就是说采用深度学习的方法可以直接估计出两帧间的运动估计，所以本文大胆地将结合深度学习的 VSLAM 归为直接法 VSLAM 中。

虽然这种直接法 VSLAM 是一个较新的方向，但是在最近大有爆发之势：CNN-SLAM[74]是比较完整的 VSLAM 系统，它使用卷积神经网络（convolutional neural networks, CNN）代替 LSDSLAM 中的深度估计和图像匹配，从单视角中得到了语义连贯的场景重建；UnDeepVO[75]采用非监督学习在训练中使用立体图像对，不仅可以估计深度和运动，而且能够构建绝对尺度的稠密深度地图；文献[76]采用 CNN 提取特征点和匹配特征点，在 CPU 上实现了实时的 SLAM，文献[77]、[78]分别利用无监督学习和监督学习完成了深度估计和运动估计，文献[79]利用 CNN 和 RNN 构建了一个（视觉惯导里程计）VIO，输入图像与惯导信息，直接输出运动，文献[80]、[81]和[82]分别利用 CNN 实现 SLAM 中的重定位功能和闭环检测模块。这些文章代表了近年来研究者们的一部分工作，虽然深度学习展示了它在 SLAM 上运动估计、重定位、闭环检测上的潜能，在速度上已经可以与传统特征点法媲美而且还有提升空间，但在精度上尚未达到 ORBSLAM 的水平。

VSLAM 是具备几何模型的优化问题，而深度学习的优势在于识别，将两者结合利用几何结构得到高精度位姿，再利用深度学习将图像与语义进行关联，生成环境的语义地图，构建环境的语义知识库[83]，这将是未来重要的发展方向。

6　发展趋势

6.1　语义 SLAM

语义 SLAM 是在传统 SLAM 的基础上构建带有标签信息的环境地图，如图 16 所示。机器人对环境的认识分为感知、认知和理解三个层面。为了让机器人具备环境理解能力，并在此基础上进行自主导航和路径规划，构建高层次的语义地图是必不可少的。语义 SLAM 有两种方式，一种是在构建完 3D 地图后进一步对地图进行语义解析，这种方法虽然精度高，但有点偏离真正的语义 SLAM；另一种方法是在估计相机位姿的同时对 2D 图像中的关键帧进行解析，再整合进 3D 图像中[84]。文献[85]使用深度图像在 ORBSLAM 框架中对每个关键帧进行目标检测和 3D 分割，然后将分割的结果进行数据关联，得到语义信息和对象实体的环境地图。文献[86]也是通过深度神经网络对深度图像进行语义分割，数据集的测试表明通过多视角一致性优化训练能够提高分割结果和系统性能。同样地，文献[87]也是对深度图像采用卷积神经网络和稠密 SLAM 系统，不仅能够生成有效的 3D 语义地图，而且能够在实时（25 Frame/S）的情况下有交互地使用。可以看出，结合语义和 SLAM 的研究还在初级阶段，目前大多是利用深度学习对稠密的 SLAM 地图进行语义上的分割，未来深度学习将在构建语义 SLAM 地图上发挥更大的作用。

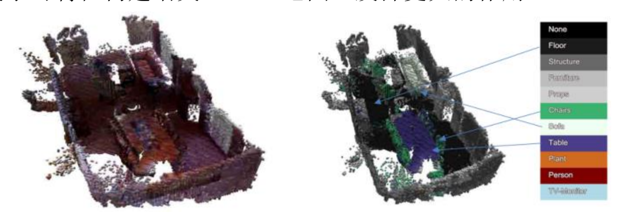


图 16　稠密地图和语义地图

Fig. 16　Dense and semantic maps

6.2　动态环境 SLAM

针对动态场景有学者做了一些探索性的工作，文献[88]提出了正态分布变换占用图（NDT-OM），结合了正态分布变换（NDT）和占用网格地图两种表示的优点，而且制定了精确的递归更新，设计了占用更新公式，在动态环境中构建一致的地图。Einhorn 等人[89]在此之上提出一种检测和处理动态物体的方法，然后结合 DNT 和占用网格地图实现基于图优化的 SLAM 算法。但是无论是精度还是实际效果还达不到需求，所以目前大多数成熟的 SLAM 方法都是假定静态环境，然后将移动部分视为异常值，但是按照人类的思维，这个模型是不对的，至少是有缺陷的。假设在一个场景中，对面有车和人（行驶或者静止），本文构建该动态地图，有如下方式：首先将地图元素进行分块包括静止物体和运动物体，一种是采用深度学习识别出建筑、树木、地面等静止物体，以及车辆行人等运动物体，通过静止物体估计相机运动，然后重构运动物体；另一种是通过人工智能方式，针对不同的物体采用不同的预测模型，如首先识别出车辆，然后识别出驾驶座上是否有人，如果没人可以当做静止物体，如果有人，会估计车子运动距离和方向，针对行人，也会估计人的运动距离和方向。与语义 SLAM 一样，动态环境的 SLAM 需要借力深度学习和人工智能，还有很大的发展空间。

6.3　多机器人 SLAM

多机器人 SLAM 有很多优点，如可执行多重任务、协同完成同一任务、执行任务耗时更短、构建地图的精度更高和容错能力更强等。多机器人 SLAM 的核心问题是多机器人之间的地图融合，如何利用共享的信息改进全局地图的精度是

关键。这有两种情况：一种是机器人之间的相对位置关系已知或者固定，只需要计算地图的转换矩阵[90]；另一种是机器人之间的相对位置未知或者变化时，可以通过各自构建的地图之间的公共区域，或者采用传感器对相对位姿进行测量，并在协同 SLAM 中作为一个待优化的边进行约束。文献[91]不知道机器人之间的相对位姿，通过匹配地图的点特征在 FastSLAM 的框架下完成地图的融合。文献[92]使用一个飞行器与一个地面机器人进行协作定位，在半结构化的室外环境中构建地图。文献[93]采用机器人独立进行 SLAM，生成独立的地图。当机器人相遇时，计算地图的相似处，进行合并生成全局地图。文献[94]提出一种飞行器和地面机器人联合定位方法，通过飞行器的机载视觉传感器对准由地面机器人的深度相机构建的地图，得到 3D 重构的稠密地图，解决了空地联合定位问题。文献[95]利用神经网络进行地图融合，它们先从网格地图中提取特征，然后根据特征计算两个地图间的旋转与平移。Zou 等人[96]专门针对动态环境开发了一款多相机的 VSLAM 系统（第一个适用于动态环境的多相机 VSLAM），该系统采用基于“intercamer”和“intracamera”的位置估计对静态点和动态点的分类，可用于多机器人相对独立地完成同时定位和建图工作。

多机器人 VSLAM 如图 17 所示，是 VSLAM 的重要部分，是实现机器人自主编队进行任务规划和导航的必需属性。未来，多机器人 VSLAM 系统框架、多机器人之间的地图融合，以及利用子地图和全局地图的重叠提高地图的精度和整体系统性能，还有如何提高传感器失效或构建地图失败后的系统容错能力（在复杂环境中尤其重要），这些都是待解决的问题和发展方向。

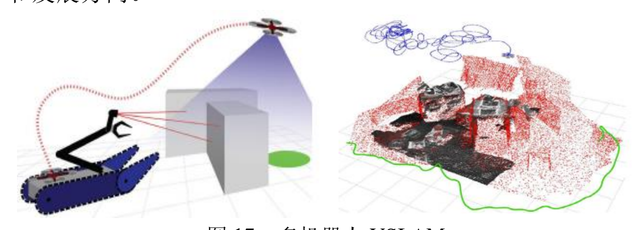


图 17 多机器人 VSLAM

Fig. 17 Multi-robot vslam

7 结束语

本文对 VSLAM 的历史和发展历程以及 VSLAM 的各个模块进行了阐述，对基于特征法、直接法和混合法的 VSLAM 技术的最新进展情况进行了分析，并详细介绍其中的关键技术包括初始化、运动跟踪及其优化算法的最新成果。经过 30 年的发展，静态环境下，VSLAM 的基本理论和系统框架已经成熟，但是动态环境和多机协同是未来 VSLAM 的必需属性也是其痛点，还需要有新的理论和新的技术给以支撑。另外，一些学者从改进 VSLAM 性能、扩展应用场景方面进行了新的尝试，如采用深度学习方法来解决图像匹配和深度估计等问题，这些均是 VSLAM 未来的发展方向，值得该领域研究者的关注。

参考文献：

[1] Durrant-Whyte H, Bailey T. Simultaneous localization and mapping: Part I [J]. IEEE Robotics & Automation Magazine, 2006, 13 (2): 99-110

[2] Bailey T, Durrant-Whyte H. Simultaneous localization and mapping (SLAM): part II [J]. IEEE Robotics & Automation Magazine, 2006, 13 (3): 108-117

[3] Liu Haomin, Zhang Guofeng, Bao Hujun. A survey of monocular simultaneous localization and mapping [J]. Journal of Computer-Aided Design & Computer Graphics, 2016.

[4] 高翔，张涛，颜沁睿，等. 视觉 SLAM 十四讲：从理论到实践[M]. 电子工业出版社，2017 (Gao Xiang, Zhang Tao, Yan Qinrui, *et al*. Visual SLAM 14 lectures: from theory to practice[M].[S.l.]: Electronic Industry Press, 2017)

[5] Hartley R, Zisserman A. Multiple view geometry in computer vision [M]. [S.l.]:Cambridge University Press, 2003.

[6] Lowe D G. Distinctive image features from scale-invariant keypoints [J]. International Journal of Computer Vision, 2004, 60 (2): 91-110.

[7] Lowe D G. Object recognition from local scale-invariant features [C]// Proc of the 17th IEEE International Conference on Computer Vision. [S.l.]:IEEE Press, 2002: 1150.

[8] Bay H, Tuytelaars T, Gool L V. Surf: speeded up robust features[C]// Proc of ECCV.2006:404-417.

[9] Rosten E, Drummond T. Machine learning for high-speed corner detection[C]// Proc of European Conference on Computer Vision. 2006: 430-443.

[10] Rublee E, Rabaud V, Konolige K, *et al.* ORB: an efficient alternative to SIFT or SURF [C]// Proc of International Conference on Computer Vision. [S.l.]:IEEE Press,2012.

[11] Siegwart R, Nourbakhsh I R, Scaramuzza D, *et al.* Introduction to autonomous mobile robots [M]. [S.l.]:MIT Press, 2011.

[12] Calonder M, Lepetit V, Strecha C, *et al.* Brief: binary robust independent elementary features [C]// Proc of European Conference on Computer Vision. Berlin:Springer, 2010: 778-792.

[13] Geiger A, Ziegler J, Stiller C. StereoScan: dense 3D reconstruction in real-time [C]// Proc of Intelligent Vehicles Symposium. [S.l.]:IEEE Press,2011: 963-968.

[14] Hartley R I. In defense of the eight-point algorithm [J]. IEEE PAMI, 1997, 19 (6): 580-593.

[15] Hartley R, Zisserman A. Multiple view geometry in computer vision [M] .[S.l.]:Cambridge University Press, 2000: 1865-1872.

[16] Li Hongdong, Hartley R. Five-point motion estimation made easy [C]// Proc of International Conference on Pattern Recognition. [S.l.]:IEEE Press,2006: 630-633.

[17] Gao Xiaoshan, Hou Xiaorong, Tang Jianliang, *et al.* Complete solution classification for the perspective-three-point problem [J]. IEEE Trans on Pattern Analysis & Machine Intelligence, 2003, 25 (8): 930-943.

[18] Lepetit V, Moreno-Noguer F, Fua P. EPnP: an accurate O (n) solution to the PnP problem [J]. International Journal of Computer Vision, 2009, 81 (2): 155-166.

[19] Penatesanchez A, Andradecetto J, Morenonoguer F. Exhaustive linearization for robust camera pose and focal length estimation [J]. IEEE Trans on Pattern Analysis & Machine Intelligence, 2013, 35 (10): 2387.

[20] Triggs B, Mclauchlan P F, Hartley R I, *et al.* Bundle adjustment:a modern synthesis [C]// Proc of International Workshop on Vision Algorithms: Theory and Practice. [S.l.]:Springer-Verlag, 1999: 298-372.

[21] Salas-Moreno R F. Dense semantic SLAM [D]. London, UK: Imperial College London, 2014.

[22] Bazeille S, Filliat D. Combining odometry and visual loopclosure detection for consistent topo-metrical mapping [J]. RAIRO:Operations Research, 2010, 44 (4): 365-377.

[23] Lowry S, Sünderhauf N, Newman P, *et al.* Visual place recognition: a survey [J]. IEEE Trans on Robotics, 2016, 32 (1): 1-19.

[24] Gálvez-López D, Tardos J D. Bags of binary words for fast place recognition in image sequences [J]. IEEE Trans on Robotics, 2012, 28 (5): 1188-1197.

[25] Botterill T, Mills S, Green R. Bag-of-words-driven, single-camera simultaneous localization and mapping [J]. Journal of Field Robotics, 2011, 28 (2): 204-226.

[26] Angeli A, Filliat D, Doncieux S, *et al.* Fast and incremental method for loop-closure detection using bags of visual words [J]. IEEE Trans on Robotics, 2008, 24 (5): 1027-1037.

[27] Cummins M, Newman P. FAB-MAP: probabilistic localization and mapping in the space of appearance [J]. The International Journal of Robotics Research, 2008, 27 (6): 647-665.

[28] Cummins M, Newman P. Accelerating FAB-MAP with concentration inequalities [J]. IEEE Trans on Robotics, 2010, 26 (6): 1042-1050.

[29] Labbé M, Michaud F. Memory management for real-time appearance-based loop closure detection [C]// Proc of IEEE/RSJ International Conference on Intelligent Robots and Systems.[S.l.]:IEEE Press, 2011: 1271-1276.

[30] Gálvez-López D, Tardos J D. Bags of binary words for fast place recognition in image sequences [J]. IEEE Trans on Robotics, 2012, 28 (5): 1188-1197.

[31] Robertson S. Understanding inverse document frequency: on theoretical arguments for IDF [J]. Journal of Documentation, 2013, 60 (5): 503-520.

[32] Nister D, Stewenius H. Robust scalable recognition with a vocabulary tree [J]. Proc CVPR, 2006, 2 (10): 2161-2168.

[33] Chow C K. Approximating discrete probability distributions with dependence trees [J]. IEEE Trans on Information Theory, 1968, 14 (3): 462-467.

[34] Dissanayake M W M G, Newman P, Clark S, *et al.* A solution to the simultaneous localization and map building (SLAM) problem [J]. IEEE Trans Ra, 2001, 17 (3): 229-241.

[35] Davison A J, Reid I D, Molton N D, *et al.* MonoSLAM: real-time single camera SLAM [J]. IEEE Trans on Pattern Analysis & Machine Intelligence, 2007, 29 (6): 1052-1067.

[36] Strasdat H, Montiel J, Davison A J. Real-time monocular slam: why filter? [C]// Proc of IEEE International Conference on Robotics and Automation . 2010:2657-2664.[S.l.]: IEEE Press, 2010.

[37] Lourakis M I A, Argyros A A. SBA: a software package for generic sparse bundle adjustment [J]. ACM Trans. Math. Softw, 2009, 36 (1): 2.

[38] Kümmerle R, Grisetti G, Strasdat H, *et al.* G2o: a general framework for graph optimization [C]// Proc of IEEE International Conference on Robotics and Automation. [S.l.]: IEEE Press,2011: 3607-3613.

[39] Agarwal S, Mierle K. Others: ceres solver.[EB/OL].https://code. google. com/p/ceres-solver/.

[40] Klein G, Murray D. Parallel tracking and mapping for small AR Workspaces[C]// Proc of the 6th IEEE and ACM International Symposium on Mixed and Augmented Reality.2007: 1-10.

[41] Nist´er D. An efficient solution to the five-point relative pose problem[J].IEEE Trans on Pattern Analysis and Machine Intelligence, 2004(26): 756-77.

[42] Faugeras O, Lustman F. Motion and structure from motion in a piecewise planar environment[J]. International Journal of Pattern Recognition and Artificial Intelligence, 1988(2): 485-508.

[43] Mur-Artal R, Montiel J M M, Tardos J D. ORB-SLAM: a versatile and accurate monocular SLAM system [J]. IEEE Trans on Robotics, 2015, 31 (5): 1147-1163.

[44] Mur-Artal R,. Tardos J D. ORB-SLAM2: an open-source SLAM system for monocular, stereo and RGB-D cameras[J]. arXiv preprint arXiv: 1610. 06475, 2016.

[45] Gálvez-López D, Tardos J D. Bags of binary words for fast place recognition in image sequences [J]. IEEE Trans on Robotics, 2012, 28 (5): 1188-1197.

[46] Strasdat H, Montiel J M M, Davison A J. Scale drift-aware large scale monocular SLAM [J]. Robotics: Science and Systems VI, 2010, 2.

[47] Mur-Artal R, Tardos J D. Probabilistic semi-dense mapping from highly accurate feature-based monocular SLAM[C]//Proc of Robotics: Science and Systems.2015.

[48] Mur-Artal R, Tardós J D. Visual-inertial monocular SLAM with map reuse [J]. IEEE Robotics and Automation Letters, 2017, 2 (2): 796-803

[49] Newcombe R A, Lovegrove S J, Davison A J. DTAM: dense tracking and mapping in real-time [C]// Proc of IEEE International Conference on Computer Vision. [S.l.]:IEEE Press, 2011: 2320-2327.

[50] Engel J, Sturm J, Cremers D. Semi-dense visual odometry for a monocular camera [C]// Proc of IEEE International Conference on Computer Vision. [S.l.]:IEEE Press, 2014: 1449-1456.

[51] Engel J, Schöps T, Cremers D. LSD-SLAM: large-scale direct monocular SLAM [C]// Proc of European Conference on Computer Vision. Cham: Springer, 2014: 834-849.

[52] Glover A , Maddern W , Warren M , *et al.* OpenFABMAP: an open source toolbox for appearance-based loop closure detection[C]// Proc of International Conference on Robotics and Automation.2012: 4730-4735.

[53] Caruso D, Engel J, Cremers D. Large-scale direct SLAM for omnidirectional cameras [C]// Proc of IEEE/RSJ International Conference on Intelligent Robots and Systems.[S.l.]: IEEE Press, 2015: 141-148.

[54] Engel J, Stückler J, Cremers D. Large-scale direct SLAM with stereo cameras [C]// Proc of IEEE/RSJ International Conference on Intelligent Robots and Systems. [S.l.]: IEEE Press, 2015: 1935-1942.

[55] Engel J, Koltun V, Cremers D. Direct sparse odometry [J]. IEEE Trans on Pattern Analysis & Machine Intelligence, 2017, PP (99): 1-1.

[56] Engel J, Usenko V, Cremers D. A photometrically calibrated benchmark for monocular visual odometry [J]. arXiv preprint arXiv: 1607. 02555, 2016.

[57] Forster C, Pizzoli M, Scaramuzza D. SVO: fast semi-direct monocular visual odometry [C]// Proc of IEEE International Conference on Robotics and Automation. [S.l.]: IEEE Press,2014: 15-22.

[58] George Vogiatzis, Carlos Hernández. Video-based, real-time multi-view stereo [J]. Image & Vision Computing, 2011, 29 (7): 434-441.

[59] Forster C, Zhang Z, Gassner M, *et al.* SVO: semidirect visual odometry for monocular and multicamera systems [J]. IEEE Trans on Robotics, 2017, 33 (2): 249-265.

[60] Forster C, Carlone L, Dellaert F, *et al.* On-manifold preintegration for real-time visual-inertial odometry [J]. IEEE Trans on Robotics, 2015, 33 (1): 1-21.

[61] Smith P. Real-time monocular slam with straight lines [C]// Proc of British Machine Vision Conference.2006: 17-26.

[62] Sola J, Vidal-Calleja T, Devy M. Undelayed initialization of line segments in monocular SLAM [C]// Proc of IEEE/RSJ International Conference on Intelligent Robots and Systems. [S.l.]:IEEE Press, 2009: 1553-1558.

[63] Eade E, Drummond T. Edge landmarks in monocular SLAM [J]. Image & Vision Computing, 2009, 27 (5): 588-596.

[64] Gomez-Ojeda R, Gonzalez-Jimenez J. Robust stereo visual odometry through a probabilistic combination of points and line segments [C]// Proc of IEEE International Conference on Robotics and Automation. [S.l.]:IEEE Press,2016: 2521-2526.

[65] Gomez-Ojeda R, Zuñiga-Noël D, Moreno F A, *et al.* PL-SLAM: a stereo slam system through the combination of points and line segments [J]. arXiv preprint arXiv: 1705. 09479, 2017.

[66] Lee J, Zhang Guoxuan, Lim J, *et al.* Place recognition using straight lines for vision-based SLAM [C]// Proc of IEEE International Conference on Robotics and Automation. [S.l.]:IEEE Press, 2013: 3799-3806.

[67] Lee Jinhan, Lee Sehyung, Lim Jongwoo, *et al.* Outdoor place recognition in urban environments using straight lines [C]// Proc of IEEE International Conference on Robotics and Automation. [S.l.]:IEEE Press, 2014: 5550-5557.

[68] Zhang Guoxuan, Lee J, Lim J, *et al.* Building a 3-D line-based map using a stereo SLAM [J]. IEEE Trans on Robotics, 2015: 1-14.

[69] Zuo Xingxing, Xie Xiaojia, Liu Yong *et al.* Robust visual SLAM with point and line features [C]// Proc of IEEE/RSJ International Conference on Intelligent Robots and Systems. [S.l.]:IEEE Press, 2017: 1775-1782.

[70] Lee G H, Fraundorfer F, Pollefeys M. MAV visual SLAM with plane constraint [C]// Proc of IEEE International Conference on Robotics and Automation. [S.l.]:IEEE Press, 2011: 3139-3144.

[71] Taguchi Y, Jian Y D, Ramalingam S, *et al.* Point-plane SLAM for hand-held 3D sensors [C]// Proc of IEEE International Conference on Robotics and Automation. [S.l.]:IEEE Press, 2013: 5182-5189.

[72] Yang Shichao, Song Yu, Kaess Michael, *et al.* Pop-up slam: Semantic monocular plane slam for low-texture environments [C]// Proc of IEEE/RSJ International Conference on Intelligent Robots and Systems. [S.l.]:IEEE Press, 2016: 1222-1229.

[73] 李海丰，胡遵河，陈新伟. PLP-SLAM: 基于点、线、面特征融合的视觉 SLAM 方法 [J]. 机器人, 2017, 39 (2): 214-220. (Li Haifeng, Hu Zunhe, Chen Xinwei. PLP-SLAM: a visual SLAM method based on point-line-plane feature fusion. [J].Robot, 2017, 39 (2): 214-220.)

[74] Tateno K, Tombari F, Laina I, *et al.* CNN-SLAM: real-time dense monocular SLAM with learned depth prediction [J]. arXiv preprint arXiv: 1704. 03489, 2017.

[75] Li Ruihao, Wang Sen, Long Zhiqiang, *et al.* UnDeepVO: monocular visual odometry through unsupervised deep learning [J]. arXiv preprint arXiv: 1709. 06841, 2017.

[76] DeTone D, Malisiewicz T, Rabinovich A. toward geometric deep SLAM [J]. arXiv preprint arXiv: 1707. 07410, 2017.

[77] Clark R, Wang Sen, Wen Hongkai, *et al.* VINet: visual-inertial odometry as a sequence-to-sequence learning problem [C]// Proc of AAAI. 2017: 3995-4001.

[78] Zhou Tinghui, Brown M, Snavely N, *et al.* Unsupervised learning of depth and ego-motion from video [J]. arXiv preprint arXiv: 1704. 07813, 2017.

[79] Vijayanarasimhan S, Ricco S, Schmid C, *et al.* SfM-net: learning of structure and motion from video [J]. arXiv preprint arXiv: 1704. 07804, 2017.

[80] Wu Jian, Ma Liwei, Hu Xiaolin. Delving deeper into convolutional neural networks for camera relocalization [C]// Proc of IEEE International Conference on Robotics and Automation. [S.l.]:IEEE Press, 2017: 5644-5651.

[81] Kendall A, Grimes M, Cipolla R. Posenet: a convolutional network for real-time 6-dof camera relocalization [C]// Proc of IEEE International Conference on Computer Vision. 2015: 2938-2946.

[82] Pillai S J. Leonard self-supervised visual place recognition learning [C]// Proc of Mobile Robots Learning for Localization and Mapping Workshop. 2017.

[83] 赵洋，刘国良，田国会，等. 基于深度学习的视觉 SLAM 综述 [J]. 机器人, 2017, 39 (6): 889-896. (Zhao Yang, Liu Guoliang, Tian Guohui, *et al.* A survey of visual slam based on deep learning[J]. Robot, 2017, 39 (6): 889-896.)

[84] Li Xuanpeng, Belaroussi R. Semi-dense 3D semantic mapping from monocular SLAM [J]. arXiv preprint arXiv: 1611. 04144, 2016.

[85] Sünderhauf N, Pham T T, Latif Y, *et al.* Meaningful maps with object-oriented semantic mapping [C]// Proc of IEEE/RSJ International Conference on Intelligent Robots and Systems. [S.l.]:IEEE Press, 2017: 5079-5085.

[86] Ma Lingni, Stückler J, Kerl C, *et al.* Multi-view deep learning for consistent semantic mapping with rgb-d cameras [C]// Proc of IEEE/RSJ International Conference on.Intelligent Robots and Systems .[S.l.]:IEEE Press, 2017: 598-605.

[87] McCormac J, Handa A, Davison A, *et al.* Semanticfusion: dense 3D semantic mapping with convolutional neural networks [C]// Proc of IEEE International Conference on Robotics and automation. [S.l.]:IEEE Press, 2017: 4628-4635.

[88] Saarinen J P, Andreasson H, Stoyanov T, *et al.* 3D normal distributions transform occupancy maps: an efficient representation for mapping in dynamic environments [J]. International Journal of Robotics Research, 2013, 32 (14): 1627-1644.

[89] Einhorn E, Gross H M. Generic NDT mapping in dynamic environments and its application for lifelong SLAM [M]. [S.l.]:North-Holland Publishing Co, 2015.

[90] Michael N, Shen S, Mohta K, *et al.* Collaborative mapping of an earthquake-damaged building via ground and aerial robots [J]. Journal of Field Robotics, 2012, 29 (5): 832-841.

[91] Lee H C, Lee S H, Choi M H, *et al.* Probabilistic map merging for multi-robot RBPF-SLAM with unknown initial poses [J]. Robotica, 2012, 30 (2): 205-220.

[92] Vidal-Calleja T A, Berger C, Solà J, *et al.* Large scale multiple robot visual mapping with heterogeneous landmarks in semi-structured terrain [J]. Robotics & Autonomous Systems, 2011, 59 (9): 654-674.

[93] Benedettelli D, Garulli A, Giannitrapani A. Cooperative SLAM using M-space representation of linear features [M]. [S.l.]:North-Holland Publishing Co. 2012.

[94] Forster C, Pizzoli M, Scaramuzza D. Air-ground localization and map augmentation using monocular dense reconstruction [C]// Proc of IEEE/RSJ International Conference on Intelligent Robots and Systems. [S.l.]:IEEE Press, 2013: 3971-3978.

[95] Saeedi S, Paull L, Trentini M, *et al.* Neural network-based multiple robot simultaneous localization and mapping [J]. IEEE Trans on Neural Networks, 2011, 22 (12): 2376-2387.

[96] Zou Danping, Tan Ping. CoSLAM: collaborative visual SLAM in dynamic environments. [J]. IEEE Trans Pattern Anal Mach Intell, 2013, 35 (2): 354-36